Deformable-Medium Affordances for Interacting with Multi-Robot Systems

Matteo Diana, Jean-Pierre de la Croix, and Magnus Egerstedt

Abstract— **This paper addresses the issue of human-swarm interactions by proposing a new set of affordances that make a multi-robot system amenable to human control. In particular, we propose to use clay – a deformable medium – as the "joy-stick" for controlling the swarm, supporting such affordances as stretching, splitting and merging, shaping, and mixing. The contribution beyond the formulation of these affordances is the coupling of an image recognition framework to decentralized control laws for the individual robots, and the developed human-swarm interaction methodology is applied to a team of mobile robots.**

I. INTRODUCTION

When it comes to controlling certain classes of systems, humans have a clear intuition about how input signals translate into motions, e.g., what the effect is when turning the steering wheel of a car. For other types of systems, this intuition becomes less clear and significant training is required to be able to operate the system successfully, as is for example the case when remotely piloting helicopters. In this paper, we investigate an untrained operator's ability to control a large collection of mobile robots.

The trend in human-controlled robotic systems is to move away from the current many-to-one paradigm, where multiple operators are required to control a single system (e.g., [11]) to a one-to-many mode of operation in which a single operator can control a large number of autonomous vehicles. For this transition to be successful, it is critical that we gain a clear understanding of how humans should interact with a large number of robots, which is precisely the human-swarm-interaction problem, e.g., [3], [9], [10].

A few different approaches to interacting with swarms of mobile robots have been previously proposed. The main line of thought is to use so-called leader-based interactions, where the user interacts directly with a small subset of the agents, e.g., [5], [12]. This is an effective strategy if the number of agents is relatively small. But, it becomes cumbersome as the number of agents increases, as was shown in [4].

Alternative approaches that have been proposed include induced flows across agents [8], boundary value control [6], or behavioral interactions [1], [2]. A related question concerns the appropriate structure of user interfaces that ensures that sufficient situational awareness is provided and that the user is not overloaded with swarm-related inputs [10], [13].

However, these approaches all start with a notion of the interaction dynamics, and then approach the human-swarm interaction problem as one concerning how this interaction dynamics can be effectively manipulated. In this paper, we take the opposite view by focusing on what constitutes effective *affordances* for interacting with large collections of mobile robots. An affordance, as defined by Gibson [7], is a quality of an object which explicitly allows a user to perform a particular action. Classic examples include a "knob" which affords twisting/pushing, or a "cord" which affords pulling.

The affordances we identify when controlling a swarm, include stretching the swarm, molding it into a particular shape, splitting and merging sub-swarms, and mixing of different swarms. One object that provides such affordances is clay, which is a physical object that it is easy to manipulate. As such, the main contribution of this paper is the identification of the clay as an effective multi-agent control interface, and the accompanying algorithms needed to identify the clay shapes, and then turn those into distributed control laws for a team of mobile robots. The outline of this paper is as follows: In Section II, we provide a high-level overview of the proposed human-swarm interaction algorithm. In Sections III and IV, we discuss the image segmentation and distributed robotics algorithms, respectively, that are used to map shapes onto control laws. The paper concludes, in Section V and VI, with experimental results, highlighting the efficacy of the proposed affordance-based framework for multi-agent interactions.

II. STRUCTURE OF THE ALGORITHM

The objective of the proposed algorithm is to provide the operator with an interface for controlling a multi-robot swarm into a particular formation by molding clay. Rather than instrumenting the clay itself, the proposed framework for interacting with the swarm is based on a combination of computer vision algorithms for monitoring the clay interface, and distributed robotics algorithms for mapping the clay shape onto the robotic platforms.

The algorithm can be divided into two phases–an image recognition phase and a swarm control phase–illustrated in Fig. 1. Whenever the operator presents a new shape (part of the shape library) by modifying the clay controller, the image recognition algorithm analyzes the observed image and produces, as result, the corresponding control law (part of the control law library). Then, this information is sent to a random agent, which starts to spread the control signal to

This work was supported through a grant from the AFOSR.

M. Diana is with the School of Information Engineering, Università di Pisa, Pisa, Italy and, as a Research Scholar, with the School of Electrical and Computer Engineering, Georgia Institute of Technology mr.matteo.diana@gmail.com

J.P. de la Croix and M. Egerstedt are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30308, USA {jdelacroix,magnus}@gatech.edu

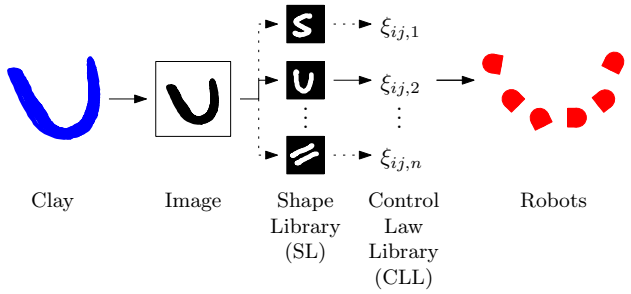


Fig. 1: Structure of the algorithm.

its neighbors. Eventually, the whole swarm will be subjected to the same control law and converge to the desired shape.

Although the main contribution of this paper is the clay-based interaction model for supporting a number of novel affordances for human-swarm interactions, this model only becomes meaningful in the context of actual algorithms. The image recognition framework (described in Section III), consists of a segmentation-inspired algorithm that classifies each observed clay shape in the image to a shape class in the precomputed Shapes Library (SL).

The distributed multi-robot control framework (described in Section IV), must map the output from the image recognition process onto executable control laws. As such, we will develop control laws such that the entire swarm can be controlled in its spatial distribution in a distributed and decentralized fashion, without being affected by the swarm size. The set of control laws forms a precomputed Control Laws Library (CLL). The SL and CLL are created off-line and between their elements there is a one-to-one correspondence, meaning that for each shape in the SL there exists a control law in the CLL.

III. IMAGE RECOGNITION FRAMEWORK

In this section we describe the image recognition algorithm needed to recognize the shape formed by the operator with the clay. The algorithm is divided into two main parts–the off-line part described in Subsections A and B and the on-line part described in Subsections C. The objective of the off-line part is to extract the features of the boundary of a shape for each class and store these features in the SL. The objective of the on-line part is to recognize the shape in an image of the clay based on the features stored in the SL.

A. Shape Alignment

For each of the m classes of shapes in the SL, we want to extract the features of the boundary of a shape (which defines that class of shapes) from a sample of images called the training set. Before we are able to extract these features, we need to apply an alignment process. The alignment process is required to remove variations in position, orientation, and scale of the shape (i.e., a shape's pose) in the different images of the training set before feature extraction. The provided raw training set consists of binary (i.e., monochrome) images of a shape from a particular class in different poses.

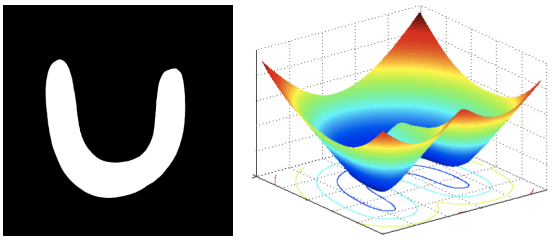
First, coarse alignment is performed by hand, and then the variational approach proposed in [15] is applied to achieve a finer alignment.

B. Parameterizing the Shape Boundaries

The features of the boundary of a shape from the k-th class can be defined by the parameterized model,

$$\Phi^k[\mathbf{w}^k] = \bar{\Phi}^k + \sum_{i=1}^{q^k} w_i^k \Phi_i^k,$$

where Φ^k is a level set function parameterized by $\mathbf{w}^k = \{w_1^k, w_2^k, \ldots, w_{q^k}^k\}$, which corresponds to the weights of q^k eigenshapes extracted through an eigenvalue decomposition of n signed distance functions and $\bar{\Phi}^k$ is the mean level set of the same n signed distance functions–described in detail in [14], [15]. Specifically, the zero level set of $\Phi^k[\mathbf{w}^k]$ describes the boundaries of the shape associated with the k-th shape class. Those boundaries are directly linked to the variability in the boundary of the shape captured by the q^k eigenshapes, meaning that by varying $\mathbf{w}^k$ it will be possible to try to match a clay shape formed by the operator.



(a) Zero level-set. (b) 3D visualization.

Fig. 2: Mean shape $\bar{\Phi}^{U-shape}$.

Returning to the U-shape class, we applied the level set methods described in [14], [15] to find $\Phi^{\text{U-shape}}$ illustrated in Fig. 2b. Figure 2a is its zero level set. We store $\Phi^{\text{U-shape}}$ in the SL and can now use it in the on-line part of the image recognition framework as way to recognize if the shape of a molded piece of clay belongs to the class of U-shapes.

C. Region-based Model For Segmentation and Recognition

The objective of the on-line part is to segment an image of molded clay and recognize its shape using all parameterized models, $\Phi[\mathbf{w}]$, stored in the SL. We use the Binary Mean Model (BMM) proposed by Yezzi et al. in [16] to compute the parameters of all $\Phi^k[\mathbf{w}^k]$ and then select the one that best segments the image. This is achieved by minimizing a cost function based on the squared difference of the ratios of the total pixel intensity to area between the inner and outer regions. The inner and outer regions are determined by the zero level set of a $\Phi^k[\mathbf{w}^k]$ applied to the image. This is illustrated for the U-shape class in Fig. 3.

The algorithm described above fits in our framework since our objective is to recognize each time what shape class (of the SL) a shape molded by the operator with the clay belongs to. Note that since the operator could change, different operators likely do not share the same manual skills when

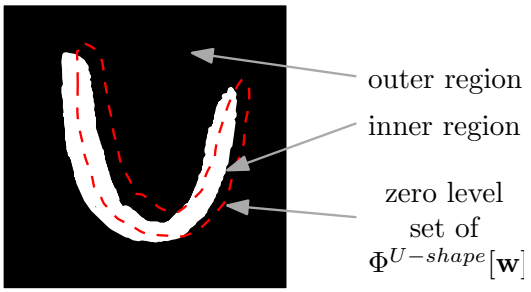


Fig. 3: A binary image of the clay with the zero level set of the current $\Phi^{\text{U-shape}}$ superimposed to demonstrate the inner and outer regions used to compute the cost in the BMM.

deforming the clay; therefore, all resulting shapes will be nonregular and nonuniform. Moreover, the segmentation has no assumptions on what shape class of the SL (i.e., which $\Phi^k[\mathbf{w}^k]$) should be used to segment the observed shape, so in a parallel fashion, our algorithm segments the new presented shape using each one of the models available in the SL.

Within each shape class k, the segmentation employed needs time to converge to its final $\Phi^k[\mathbf{w}^k]$. In our case the goal is to perform on-line shape recognition and, hence, we allow the computation of only few updates of the weight parameters $\mathbf{w}^k$ for $k = 1, 2, \ldots, m$. Subsequently, we evaluate the BMM cost functional that produced the last $\Phi^k[\mathbf{w}^k]$, for $k = 1, \ldots, m$. The smallest value across the cost functionals will be selected and, thus, determine to which shape class the shape belongs.

For example, suppose that the operator molds a U-like shape and that the precomputed SL consists of U-shape and Line-shape classes. The developed algorithm will try to segment the observed U-like shape using both classes. Since the models in the SL capture the characteristics of a particular class of shapes, when the segmentation starts using $\Phi^{U-shape}$, it will result in greater decrease in the BMM cost functional than using $\Phi^{Line-shape}$ for the segmentation. The proposed algorithm handles pose differences in the observed images by matching the latter with the corresponding parameterized shape models after the recognition phase.

IV. DISTRIBUTED SWARM CONTROL FRAMEWORK

In this section we describe how, through local information in a distributed and decentralized manner, it is possible to control large-scale swarms. It is assumed that agents do not share a global reference system, they do not have a unique ID, and they act in an asynchronous and independent fashion. Each one of the following control laws will be chosen based on the result of the image recognition algorithm (described in Section III). The selected control law will be executed until the swarm converges to the desired spatial distribution that corresponds to the shape of the clay.

A. Multi-agent Network

Let $r_i(t) \in \mathbb{R}^s$ for $i = 1, 2, \ldots, N$, where N is the number of agents, be the state of agent i at the time t. We assume that the interaction dynamics are ruled by pairwise interactions, i.e., if agents i and j are connected, then they share relative state information. Each agent is designed to maintain the desired distance and orientation with respect to its neighbors. As we will show, we can prove that the swarm will converge to the desired spatial distributions under our control laws based on existing results in algebraic graph theory. In a graph representation, the agents are described by nodes $V = \{v_1, v_2, \ldots, v_N\}$ and the connections between agents become edges $E \subseteq V \times V$, where the cardinality of E represents the number of edges. We consider two underlying graphs, namely the communication graph $G_{comm} = (V, E_{comm})$ and the sensing graph $G_{sens} = (V, E_{sens})$. We assume that these underlying graphs of the network are undirected. Let $R_{comm} > R > 0$ be the communication and sensing ranges, respectively. In our case, some control laws could cause topology variations of G_{sens}. However, such variations do not affect G_{comm} which is static and remains connected all times.

B. Agent Dynamics

To define the agents' dynamics, we use general, energy-based definitions [14], which allow agents to achieve distance-based formations. Consider a group of N mobile agents moving in $\mathbb{R}^2$, with dynamics expressed by a single integrator,

$$\dot{r}_i = u_i, \qquad i = 1, 2, , \ldots, N \tag{1}$$

where $r_i = (x_i, y_i)^T$ is the position of agent i and $u_i = (u_{ix}, u_{iy})^T$ is its input. Let $\|r_{ij}\|$ be the distance between agents i and j. For each edge j incident to agent i we define a nonnegative potential function ξ_{ij} dependent on the distance and orientation between i and j such that,

- ξ_{ij} has a unique minimum,
- ξ_{ij} is monotonically increasing near $\|r_{ij}\| = R$.

Such edge-tension energy can be defined as

$$\xi(r) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \xi_{ij}(r_i(t), r_j(t)) \tag{2}$$

which is the summation of the local contributions

$$\xi_{ij}(r_i(t), r_j(t)) = \begin{cases} \frac{1}{2}\{e_{ij}(\|r_{ij}\|)\}^2 & (v_i, v_j) \in E_{sens} \\ 0 & (v_i, v_j) \notin E_{sens} \end{cases}$$

where $e_{ij} : \mathbb{R}^+ \to \mathbb{R}$ is a strictly increasing function. Let d_{ij} be a parameter that expresses the desired interagent distance and orientation. Then, $e_{ij}(d_{ij}) = 0$. Each agent tries to minimize the edge-tension energy (2) through gradient descent:

$$u_i = -\sum_{j \in N_s(i)} \frac{\partial \xi_{ij}(r_i(t), r_j(t))^T}{\partial r_i}, \qquad i = 1, 2, , \ldots, N \tag{3}$$

where $N_s(i) = \{j | (i, j) \in E_{sens}\} \subseteq V \setminus \{i\}$ is the neighboring set of agent i. We now restate the problem in a more useful way based on the definitions made so far. By stacking the position vectors of (1), we can derive a different dynamical system. This system has $\bar{r}$ as its state, where $\bar{r} = (B_{K_N} \otimes I)\, r$ is the stack vector of all relative positions between agents, B_{K_N} is the oriented incidence matrix of the complete communication graph with N vertices, K_N represents an arbitrary orientation, $\otimes$ denotes the Kronecker matrix product, I is the

identity matrix of appropriate dimension, and r is the stack vector of agent positions. In our case the dynamics can be expressed

$$\dot{\bar{r}} = (B_{K_N} \otimes I_2)u, \tag{4}$$

where u is the stack vector of all inputs defined in (3). Since G_{comm} remains connected for all time, the communication links are fixed and time invariant.

C. Swarm Energies

Next, we show how different edge-tension energies modify the swarm spatial distribution by means of the local control laws defined in (3). These control laws form the Control Laws Library (CLL). Under the assumption that the agents' initial positions are not all coincident, but clustered closely, proofs of convergence will be given. Let θ_{ij} be the angle under which agent i sees agent j and let $d_{i,loc}$ be the distance reached so far by agent i with respect to its (local) initial position.

1) Stretching: An example of e_{ij} that allows the swarm distribution to be stretched in the preferred direction and orientation is

$$e_{st,ij} = \|r_{ij}\|^2 - \big(\cos(\theta_{ij} - \theta_{ref})\Delta + \sin(\theta_{ij} - \theta_{ref})\delta)\big)^2, \tag{5}$$

where the quantities Δ and δ represent the stretching factors with respect to the reference orientation θ_{ref}. If $\Delta = \delta$ then the swarm formation is a circle, while when $\Delta \neq \delta$ then the swarm is stretched into an ellipsoid, which is stretched most in the direction of the larger of these two "gains".

***Theorem** 1 (**Stretching**): Consider a system of N mobile agents with dynamics (1), each steered by control law (3) with local energy as in (5). Then the system approaches a configuration that minimizes all agents' energy potentials and all pairwise agents satisfy the desired interagent positions and orientations.*

Proof: Consider the function ξ defined in (2) that is differentiable and continuous everywhere[1]. The communication graph diameter cannot be larger than $(N-1)$ since the graph is connected. This implies that the largest distance between any two agents is smaller than $(N-1)R_{comm}$. Moreover, the sum of the interagent distances is bounded, such that $\sum_{(i,j)\in V\times V}\|r_{ij}\| \leq N(N-1)^2 R_{comm}/2$. Therefore, $\bar{r}$ always evolves in a closed and bounded set. Likewise, the level sets of ξ define compact sets and are bounded. From connectivity we know that a path connecting nodes i and j has length at most $(N-1)$. Let $a > 0$, from the properties of ξ we have $\|r_{ij}\| \leq \xi_{ij}^{-1}(a(N-1))$. So, the set

$$\Omega = \left\{ (\bar{r}) \,|\, \|\bar{r}\| \leq \frac{N(N-1)^2 R_{comm}}{2} \right\} \tag{6}$$

is compact.

We now show that Ω is nonincreasing and establish the invariant properties of Ω. Consider $\partial_{r_{ij}}\xi_{ij} = \partial_{ri}\xi_{ij} = -\partial_{r_j}\xi_{ij}$ and take the derivative of ξ, then it follows that,

$$\begin{aligned}\dot{\xi} &= \frac{1}{2}\sum_{i=1}^{N}\dot{\xi}_i = \frac{1}{2}2\sum_{i=1}^{N}\dot{r}_i^T \sum_{j\in N_{s(i)}}\partial_{ij}\xi_{ij} \\ &= \sum_{i=1}^{N}(\partial_{r_i}\xi_i)^T\dot{r}_i = -\sum_{i=1}^{N}(\partial_{r_i}\xi_i)^T(\partial_{r_i}\xi_i).\end{aligned} \tag{7}$$

The last term in (7) can be alternatively rewritten as $-\sum_{i=1}^{N}\|\partial_{r_i}\xi_i\|^2 \leq 0$ which is negative semidefinite and zero whenever the argument of the sum is zero, $\forall i$. Let $t = \big(\cos(\theta_{ij} - \theta_{ref})\Delta + \sin(\theta_{ij} - \theta_{ref})\delta)\big)$, then the zeros of $\dot{\xi}$ are those satisfying $\sum_{j\in N_s(i)} e_{st,ij}(r_i - r_j) = 0, \forall i = 1,\ldots,N$ so either $r_i = r_j$ or $e_{st,ij} = \big((x_i - x_j)^2 - (y_i - y_j)^2\big) - t^2 = 0$. The latter represent a circle centered in agent j's position with the radius function of the desired interagent distance and orientation of the pairwise to which the agent i should belong. Let the set $\gamma = \big\{r_i \,|\, -\sum_{i=1}^{N}\|\partial_{r_i}\xi_i\|^2 = 0\big\} \subset \Omega$. Applying LaSalle's invariance principle from initial conditions in Ω, the solutions of the system converge to a subset of γ, more precisely of $\{r_i | x_i, y_i \in span\{\gamma\}\}$. If that, agents' relative velocities $\dot{r}_{ij} = 0$ and the system dynamics becomes $\dot{\bar{r}} = -(B_{K_N}\otimes I_2)[\ldots\partial_{r_{ij}}\xi_{ij}\ldots]^\top$. So both $\dot{x}_i, \dot{y}_i$ belong to the range of the oriented incidence matrix B of G_{comm}. For a connected communication graph, $range(B_{K_N}) = span(\gamma)^\perp$. Thus, in the invariant set within γ, $x_i, y_i \in span\{\gamma\}$ and this implies that $\dot{x}_i, \dot{y}_i \in span\{\gamma\}$. The latter result allows to conclude that we are faced with a contradiction unless $\dot{x}_i, \dot{y}_i \in span\{\gamma\} \cap span\{\gamma\}^\perp \equiv \{0\}$. In the end, the control law (3) is zero at steady state implying ξ_{ij} is locally minimized. If ξ_{ij} is (locally) convex within the communication range, then the extremum is unique and the agents are stabilized to their desired positions and orientations. Collision avoidance is not ensured by the definition of ξ_{ij} so an appropriate algorithm that provides this has been implemented. ■

The existence and uniqueness of the solutions of (4) is provided by the boundness of (3). If one of the two assumptions (connectedness of G_{comm} and initial positions not all coincident, but clustered closely), cannot be guaranteed, it could happen though, that $r_{ij} \notin \Omega$ and, by consequence, the stability is not guaranteed.

We have intentionally excluded both the theorem statements and the proofs for the following swarm energies, since they follow in the same fashion as Theorem **1**.

2) Bending: A series of different spatial distributions that represent curve shapes were obtained by other choices of e_{ij}, e.g., U-shapes, S-Shapes, L-Shapes rotated by some arbitrary angle. One such potential is

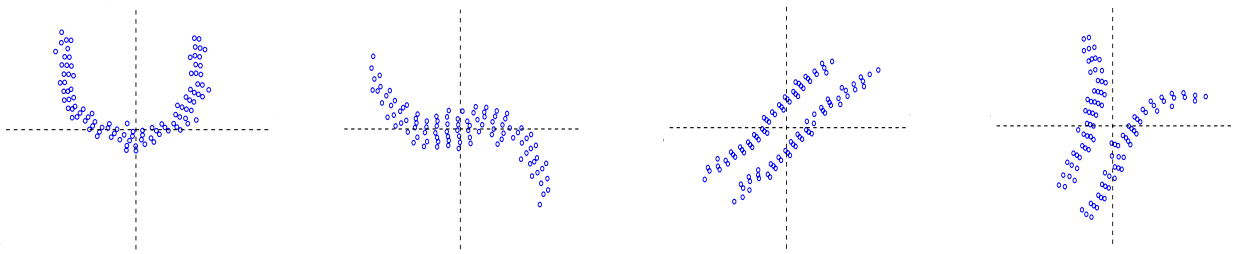
$$e_{bend,ij} = \|r_{ij}\|^2 - (\cos(\theta_{ij} - \lambda)\Delta + \sin(\theta_{ij} - \lambda)\delta))^2 \tag{8}$$

where

$$\begin{cases} \lambda = (\theta_{ref} + \sigma) \\ \sigma = \pm f(\|d_{i,loc}\|)\frac{\pi}{c} \end{cases}$$

with $c > 0$ constant. Let $sec* > 0$ be circular sections of progressively increasing distances then $f(\|d_{i,loc}\|) = \{1, 2, \ldots, n\}$ if $\|d_{i,loc}\|$ belongs to $\{sec1, sec2\ldots, secn\}$ respectively. The effect of λ is to modify the orientation angle

[1]Due to discrete variations in the neighboring sets, some discontinuities could arise in the (local) control laws.



(a) U-shape: $\theta_{ref} = 0$. (b) S-shape: $\theta_{ref} = 0$. (c) Split and stretch shape: $\theta_{ref} = \frac{\pi}{4}$. (d) Split and bend shape: $\theta_{ref} = \frac{\pi}{3}$.

Fig. 4: Spatial distributions.

of each pairwise depending upon which section an agent belongs to.

3) Splitting and Organizing: The purpose of the following is to provide the (decentralized and distributed) splitting capability to the swarm. Then, the subgroups could arrange themselves in different spatial distributions using one of the edge-tension energies described above. Once the swarm separates, it could happen that G_{sens} is no longer connected. Nevertheless, it is still possible to ensure convergence of the system through the control law of the form in (3), since G_{comm} remains connected. The local contributions to the edge-tension energy in (2) have to be slightly modified to capture these features

$$\xi_{split,ij} = \begin{cases} \pm\frac{1}{2} r_i^2 \, k \, \text{sign}[\cos(\rho) \;\; \sin(\rho)], & \|d_{i,loc}\| \leq sec1 \\ \xi_{ij}, & otherwise \end{cases}$$

where $\rho = \theta_{ref} - \frac{\pi}{2}$ and $k > 0$ is a design parameter. This particular choice of ρ splits the swarm in the perpendicular direction with respect to the desired orientation θ_{ref}.

4) Merging: For the sake of completeness, an appropriate strategy has been developed to allow the swarm to rejoin if it has been split. The basic idea of merging relies on the possibility to drive all agents to locally agree on their (local) initial positions, by applying a self edge-tension energy such as

$$\xi_{i,loc} = \frac{1}{2}(\|d_{i,loc}\| - p)^2$$

where $p \geq 0$ is the desired distance on which agree, all agents will eventually return to their initial positions.

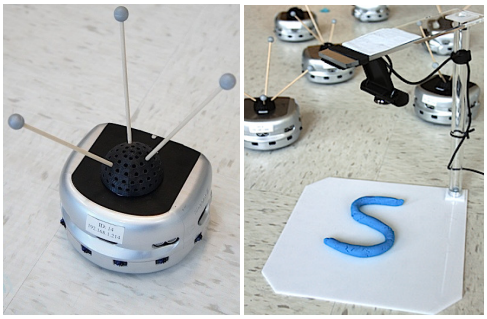
V. NUMERICAL SIMULATIONS

In this section we verify numerically the stability results obtained in Section IV in the MATLAB environment. In all simulation examples, the swarm consists of one hundred mobile agents with identical first order integrator dynamics. Initial positions were generated by an uniform distribution in an area of 10×10 units, centered at the origin. In the simulations, the single agent is assumed to be dimensionless, although the collision avoidance algorithm ensures collision free movements. Simulations verify that the system converges to invariant sets that correspond to the desired formations obtained by the energy functions described in Section IV. Extended large-scale simulations have been performed and along with the simulation parameters $\Delta = 8$, $\delta = 0.5$, sensing range $R = 6$, some examples of such formations are depicted in Fig. 4. The recognition algorithm requires less than 3 seconds to observe and recognize the presented image on a PC equipped with a 2 Ghz Intel Core 2 Duo CPU and 4 GB 1067 Mhz DDR3 RAM memory.

The case of heterogeneous swarms has also been tested and some fascinating behaviors emerged. Consider the case where there are two types of robots with different sensing ranges. Using the same energy functions defined in Section IV, the simulations showed how agents with bigger sensing range tend to move themselves to the extremities of the formation due to the increased cardinality of the neighboring set with respect to agents with smaller sensing range.

VI. EXPERIMENTAL RESULTS

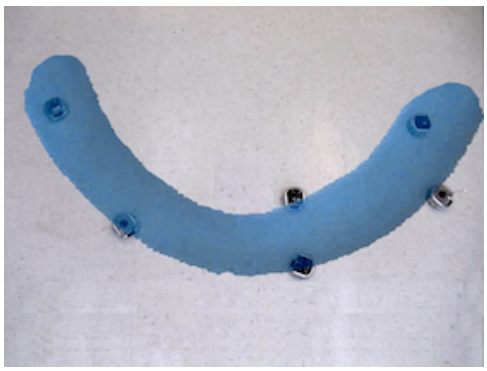
In addition to mathematical modeling and computer simulations, in this section we provide a brief description of the experimental set up and the results obtained. All experiments were conducted using a group of Khepera III robots, Fig. 5a. A camera was computer-controlled, and positioned above a high contrast surface where the clay had to be placed, Fig. 5b. Each time a new shape has been detected the image recognition algorithm computed the appropriate match from the SL to the CLL. Once the PC has computed the appropriate control law, the signal has been sent through



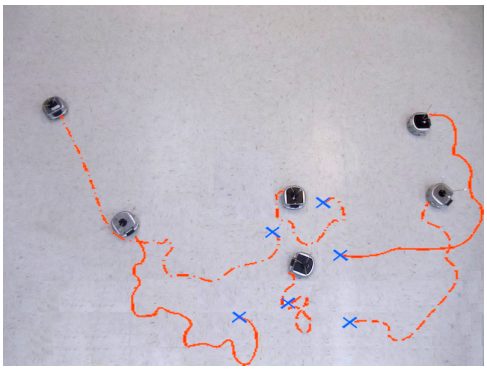
(a) Khepera III. (b) Experimental set up.

Fig. 5: Experimental elements.

WiFi to a random agent in the swarm. Based on the properties of G_{comm} the signal will be spread out to the whole swarm. The result for the U-shape in the case of six robots is depicted in Fig. 6a and the parameters in the experiment are $\Delta = 0.9$, $\delta = 0.13$, *sensing range* $R = 1$. The figure shows how the swarm successfully reproduced the desired shape. The trajectories of the robots were also measured and are



(a) U-shape superimposed on the swarm: $\theta_{ref} = 0$.



(b) Trajectories of the swarm.

Fig. 6: Experimental results

shown in Fig. 6b. The crosses in the figure represent the agents' initial positions.

The agents' positions have been obtained by an appropriate optical motion capture and tracking system. Several limitations affected the experiments with real robots: the number of robots, the model uncertainties and the transformation between simulations to Kheperas III dynamics were the major ones.

It is important to note that we were not trying to match the exact shape in the SL, but for example, we were doing a generic U-shape instead, since the parameters of the shape controllers in the CLL were previously determined off-line. Control laws acted locally and dependent on the initial positions. In Fig. 7 are depicted some U-like shapes obtained and although all formations are different from one other, the swarm successfully converged to an U-like shape each time, much like in the submitted video.

Regarding the convergence time, the proposed method has been tested with different number of robots and the larger is the group, the greater the convergence time. For the case of six robots, the convergence times vary from 20 to 35 seconds depending on the shape that has to be performed.

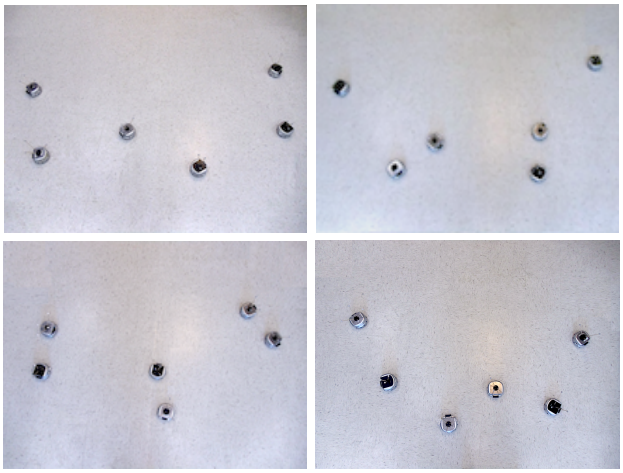


Fig. 7: U-like shapes

REFERENCES

[1] R. Arkin and K. Ali. Integration of reactive and telerobotic control in multi- agent robotic systems. *Proc. Third International Conference on Simulation of Adaptive Behavior*, pages 473-478, 1994.

[2] A. Atherton and M. Goodrich. Supporting Remote and Mobile Manipulation with an Ecological Augmented Virtuality Interface, *Proceedings of AISB-HRI Symposium New Frontiers in Human-Robot Interaction*, Scotland, 2009.

[3] M. Cummings. Human Supervisory Control of Swarming Networks. *2nd Annual Swarming: Autonomous Intelligent Networked Systems Conference*, 2004.

[4] J.P. de la Croix and M. Egerstedt. Controllability Characterizations of Leader-Based Swarm Interactions. *AAAI Symposium on Human Control of Bio-Inspired Swarms*, Arlington, DC, Nov. 2012.

[5] J. Fax, and R. Murra. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):14651476, 2004.

[6] N. Ghods and M. Krstic. Multiagent deployment over a source, IEEE Transactions on Control Systems Technology, vol. 20, pp. 277-285, 2012.

[7] J.J. Gibson. *The Theory of Affordances. In Perceiving, Acting, and Knowing*, Eds. Robert Shaw and John Bransford, 1977.

[8] P. Kingston and M. Egerstedt. Distributed-Infrastructure Multi-Robot Routing using a Helmholtz-Hodge Decomposition. *IEEE Conference on Decision and Control*, Orlando, FL, Dec. 2011.

[9] Z. Kira and M. Potter. Exerting human control over decentralized robot swarms. In *4th International Conference on Autonomous Robots and Agents (ICARA)*, pp. 566 571, 2009.,

[10] J. McLurkin, J. Smith, J. Frankel, D. Sotkowitz, D. Blau, and B. Schmidt. Speaking swarmish: Human- robot interface design for large swarms of autonomous mobile robots. *AAAI Spring Symposium*, 2006.

[11] B. Mekdeci and M.L. Cummings. Modeling Multiple Human Operators in the Supervisory Control of Heterogeneous Unmanned Vehicles. *9th Conference on Performance Metrics for Intelligent Systems*, 2009.

[12] M. Mesbahi and M. Egerstedt, Graph Theoretic Methods in Multiagent Networks, Princeton University Press, NJ, 2010.

[13] E. Olson, J. Strom, R. Morton, A. Richardson, P. Ranganathan, R. Goeddel, M. Bulic, J. Crossman, and B. Marinier. Progress towards multi-robot reconnaissance and the MAGIC 2010 Competition, *Journal of Field Robotics*, Vol. 29, No. 5, pp. 762–792, 2012.

[14] S. Osher and J. Sethian, Fronts propagation with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations, J. Comput. Phys., vol. 79, pp. 1249, 1988.

[15] A. Tsai, A. J. Yezzi, and A. S. Willsky. A shape-based approach to the segmentation of medical imagery using level sets. IEEE Trans. on Medical Imaging, 22(2):137154, 2003.

[16] A. Yezzi, A. Tsai, and A. Willsky, A statistical approach to snakes for bimodal and trimodal imagery, in Proc. Int. Conf. Computer Vision, vol.2, 1999, pp. 898903.